Combining Document Representations for Known-Item Search

Paul Ogilvie and Jamie Callan
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
pto@cs.cmu.edu, callan@cs.cmu.edu

ABSTRACT

This paper investigates the pre-conditions for successful combination of document representations formed from structural markup for the task of known-item search. As this task is very similar to work in meta-search and data fusion, we adapt several hypotheses from those research areas and investigate them in this context. To investigate these hypotheses, we present a mixture-based language model and also examine many of the current meta-search algorithms. We find that compatible output from systems is important for successful combination of document representations. We also demonstrate that combining low performing document representations can improve performance, but not consistently. We find that the techniques best suited for this task are robust to the inclusion of poorly performing document representations. We also explore the role of variance of results across systems and its impact on the performance of fusion, with the surprising result that the correct documents have higher variance across document representations than highly ranking incorrect documents.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *retrieval models*

General Terms

Algorithms, Experimentation

Keywords

Language models, known-item finding, meta-search algorithms, data fusion

1. INTRODUCTION

Known-item finding is an important information seeking activity that has recently gained some attention in the information retrieval community. In this task the user knows of a particular document, but does not know where it is. Example document types may be a web page, a report, or a normal text document. Very often, these documents have structural markup, such as HTML. We believe that by forming a variety of document representations using this structural information and combining these representations during retrieval, systems can improve retrieval performance over using the single best document representation.

One natural approach to combining document representations can be borrowed directly from the meta-search problem. The goal in meta-search is to combine the results from different search engines to produce a single ranked list that is better than the results of any single search engine. This is sometimes referred to as data fusion. Combining document representations is similar to creating a search engine for each of the document representations and performing meta-search to combine the result lists.

An alternative to meta-search techniques is to use the document representations to modify term weights directly within a single search engine. We present a technique for this using generative language modeling. Rather than using the language model estimated from a single document representation, this approach estimates a mixture language model based on a combination of language models created from the various document representations.

There has been extensive work on studying the effectiveness of meta-search and the conditions for success [5][14][19]. Aslam and Montague [1] summarize this work by stating:

> "The systems being combined should (1) have compatible output (e.g., on the same scale), (2) each produce accurate estimates of relevance, and (3) be independent of each other."

This paper investigates whether these hypotheses should also extend to the task of combining document representations for known-item finding.

The first hypothesis can be directly applied to known-item finding. We investigate this hypothesis of score compatibility using Okapi and language modeling retrieval systems across different document representations. Croft [5] describes score compatibility as systems having "comparable output in that they are trying to make the same decision within the same framework". Within Okapi and language modeling systems, Croft's statements hold true when combining document representations. However, it is difficult to recover the probability estimates from the ranking function used by Okapi. This may introduce some incompatibilities across the document representations. To

SIGIR'03, July 28–August 1, 2003, Toronto, Canada.

investigate this, we define a measure of compatibility that compares the shape of the normalized ranking functions across document representations.

The second hypothesis does not directly apply, as the notion of relevance used in ad-hoc retrieval is different from the goal of known-item finding. In known-item finding the goal is not to exhaustively find documents about a topic, but to find a single correct document. A more appropriate hypothesis in this setting would be that the document representations tend to give higher weight to the correct document than to incorrect documents. Even so, we would like an approach that is robust to the inclusion of representations that only sometimes give high score to the correct document. For example, a document representation of image alternate text in HTML may do very well sometimes at finding the correct document, but will tend to do rather poorly in general. The ideal approach would be robust to errors in the poorly performing representations, but would also be able to leverage the representation when the correct answer is found. We investigate the quality of representation hypothesis and also investigate whether any of the approaches can gain from poorly performing document representations.

The third hypothesis states that the systems should be independent of each other. That is, the systems should give high scores different sets of non-relevant documents than each other, but there should be a higher correlation among the relevant documents. Applied to known-item finding, an appropriate hypothesis is that the representations would give widely varying scores for the incorrect documents, but tend to score the correct document highly, with lower variance.

In this paper, we explore these hypotheses using both meta-search algorithms and an approach motivated by language modeling. Section 2 describes related work and the techniques used in this paper. Section 3 describes our experimental methodology, evaluation, and system details. In Section 4 we present experimental results. We conclude the paper in Section 5.

2. COMBINING REPRESENTATIONS

There are two ways to combine document representations in a retrieval system. A meta-search approach would treat each document representation as a unique search engine possibly using different search algorithms, and then combine retrieval results from the separate engines to produce one ranked list. A language modeling approach would combine the document representations into one mixture language model that estimates the query generation process, and then perform retrieval using the mixture language model. Merits of the meta-search approach include relaxed assumptions about the compatibility of the systems performing retrieval on the different document representations. With the mixture-based language modeling approach we have guidance on how to combine the probability distributions. The mixture-based language model also combines evidence at the query term level, rather than after performing retrieval.

Previous experiments in combining document representations include experiments with controlled vocabularies, passages, and phrases. Croft [5] provides an overview of these methods and experiments. Of particular interest here are experiments in the use of citations, which led to the use of link text and additional structural information on the web [3][4][9][22]. These experiments using link text all found that the link text was beneficial for homepage searching or site finding. The work reported here investigates additional document representations for known-item searching on the web.

2.1 Meta-Search/Data Fusion

Meta-search fusion algorithms were developed to address the combination of retrieval results from many retrieval systems. In general, they combine the result lists containing the top n documents from each retrieval system. This subsection describes the meta-search algorithms to which we will compare the language model approach, and discusses other related work where appropriate.

There are two main classes of meta-search fusion algorithms: ones that use scores from systems and ones that do not. The algorithms investigated in this paper are combMNZ, combSUM, Condorcet fuse, Borda fuse, and a reciprocal rank fusion strategy. Montague and Aslam compare most of these approaches for meta-search in [1] and [13]. Within these classes are variants that do and do not use training data.

Condorcet fuse, Borda fuse, and the reciprocal rank fusion strategy do not use scores from the systems. These approaches assume that the scores from the different retrieval systems are not directly comparable. In a meta-search environment, this can be a good assumption, as the different search systems may use different ranking formulas or have different corpus statistics. The corpus statistics may still be different in document representation fusion, but we can assume that the search engines use the same ranking function. This may or may not lead to comparable scores, depending on the ranking function.

Condorcet fuse [13] and Borda fuse [1] were originally developed to address elections and have been adapted to meta-search fusion. We do not discuss the Condorcet fuse algorithm in detail here, as it is beyond the scope of this paper. Borda fuse sums n – the rank of the document across all systems and sorts the documents in descending order (documents with higher scores are higher in the merged list). System weights can be incorporated by multiplying the weight by the scores for the documents. The reciprocal rank strategy [22] sums one over the rank the document across all search engines. Documents are sorted in descending order. The reciprocal rank strategy gives much higher weight than Borda fuse to documents that occur near the top of a list. System weights can be incorporated by multiplying each document's inverse rank by the weight.

Both combMNZ and combSUM are variants of an algorithm developed by Fox [6]. combSUM ranks each document using the sum of the scores returned from the individual retrieval systems, and combMNZ ranks by the sum multiplied by the number of systems that returned the document was in the top n results. It is common to perform a linear normalization of the scores in the result lists for both algorithms so that the first document in each list has a score of one and the n^{th} document has a score of zero.

We introduce additional variant of these algorithms in which this the exponential function (e^{score}) is applied to scores before the optional linear normalization. This transformation is justified when the retrieval system returns the log of the query generation probability, because this places the scores back on the probability scale. However, we will also consider this approach for as an ad-

Table 1: Testbed characteristics

Task	Corpus	Number of Topics	Number of Documents	Size	Document types
Homepage finding	WT10G	145	1,692,096	10 GB	html
Named-page finding	.GOV	150	1,247,753	18 GB	html, doc, pdf, ps

hoc normalization of Okapi scores. Weights can be incorporated by multiplying the (normalized) document scores by the weight.

Weighted combSUM is similar to using a linear combination [19] of the scores, but only uses scores in the top n results from each system. Other approaches not evaluated in this paper include a logistic regression model [18] and an approach that models score distributions [12].

2.2 Combining Language Models

A unigram language model defines a multinomial probability distribution over all words in the vocabulary of the corpus. These probabilities are interpreted as word generation probabilities, and documents are ranked by their probability of generating the query [3][16][20]. This generation probability is computed by taking the product over all query terms of the probability of the query term given the language model:

$$P(Q|\theta_D) = \prod_{i=1}^{|Q|} P(q_i|\theta_D) \tag{1}$$

where q_i is the i^{th} query term of query Q, $|Q|$ is the length of Q, and θ_D is a language model estimated from document D. In typical language modeling experiments for information retrieval, θ_D is estimated using a linear interpolation of a language model estimated from the document text and one from the entire corpus:

$$P(w|\theta_D) = \lambda_1 P_{MLE}(w|D) + \lambda_2 P_{MLE}(w|C) \tag{2}$$

where C is the entire collection. These language models are estimated using the maximum likelihood estimate (MLE) of the multinomial distribution. In this case, it is the same as the empirical distribution. The MLE estimate for a document is defined as:

$$P_{MLE}(w|D) = \frac{count(w,D)}{|D|} \tag{3}$$

The MLE distribution for the collection is estimated similarly. It is common to set the linear interpolation parameters λ_1 and λ_2 using guidance from Dirichlet prior smoothing [20][21]:

$$\lambda_1 = \frac{|D|}{|D| + \mu}, \quad \lambda_2 = \frac{\mu}{|D| + \mu} \tag{4}$$

where μ is a parameter often set near the average document length in the collection.

However, we wish to explore the combination of several language models estimated from different document representations. One approach to take for combine language models created from different document representations is linear interpolation:

$$P(w|\theta_D) = \sum_{i=1}^{k} \lambda_i P(w|\theta_{D(i)}) \tag{5}$$

where k is the number of language models, $D(i)$ is the document's i^{th} representation, and λ_i is the weight on the model $\theta_{D(i)}$. To ensure that this is a valid probability distribution, we must place these constraints on the lambdas:

$$\sum_{i=1}^{k} \lambda_i = 1 \quad \text{and for } 1 \le i \le k,\ \lambda_i \ge 0 \tag{6}$$

The form of linear interpolation presented in Equation 2 is a special case where k=2. The linear interpolation parameters can be trained or hand-tuned to the task.

We estimate the probability distribution for each model $\theta_{D(i)}$ by taking a linear interpolation of the MLE of the text observed in the i^{th} document representation and a collection language model estimated from all document representations of the same type in the collection. We set the linear interpolation weights according to Dirichlet prior smoothing. This is the model we used in [3].

2.3 Related Work

The main difference between the language modeling approach presented here and the meta-search techniques is that it combines the document representations on the query term level, rather than as a post-retrieval score combination. The approach presented here is not unique in that characteristic. For example, much recent work in the Initiative for Evaluation of XML retrieval [7] combines document components for document retrieval. Some methods being investigated in this context combine vectors or probability distributions of document components. While the document components are organized hierarchically, they could be easily adapted to the problem of combining document representations.

There are other examples of research where document representations are combined within a model. One related approach was proposed for language models in [9], but was not implemented. Myaeng et al [11] combine terms found in different document representations using Bayesian inference networks. The approach allows for different weights on terms found in different document representations, similar to what is done in the language model approach presented here. The authors had some success with this approach on limited ad-hoc retrieval experiments within a subset of the Patent data in the TREC document collection.

3. EXPERIMENTAL METHODS

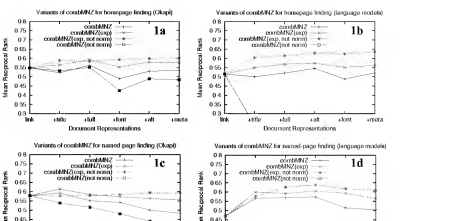
We performed experiments on two TREC tasks and corpora. Table 1 summarizes the corpora and test topics. Our experiments were on the Homepage Finding task from TREC 10 and the Named-Page Finding task from TREC 11. The Homepage Finding task has 100 training topics, while the Named-Page Finding task has no training data. The Homepage Finding task uses the WT10G corpus, and the Named-Page Finding task uses .GOV corpus. As there is no training data for the Named-Page Finding task, we only present the results using equal weights on the document representations for this task.

Table 2: Performance of individual document representations using Okapi

	Homepage		Named-page	
	MRR	# by 10	MRR	# by 10
FULL	0.239	69	0.578	112
In-LINK	0.548	94	0.438	85
TITLE	0.345	81	0.371	74
ALT	0.141	32	0.158	34
FONT	0.164	40	0.146	34
META	0.067	14	0.107	21

Table 3: Performance of individual document representations using language models

	Homepage		Named-page	
	MRR	# by 10	MRR	# by 10
FULL	0.300	77	0.469	100
In-LINK	0.515	95	0.455	87
TITLE	0.332	82	0.406	84
ALT	0.186	35	0.194	42
FONT	0.155	44	0.191	38
META	0.115	20	0.144	32



Figures 1a-d: Variants of combMNZ for combining document representations. Applying the exponential function improved performance for all tasks and systems.

To evaluate the Homepage Finding task and the Named-Page Finding task, we use mean-reciprocal rank (MRR) and number of topics where the correct page was found by rank 10 [8].

Our experiments are performed using the Lemur toolkit [10]. A separate index is created for each document representation. We use the Porter stemmer and InQuery's stopword list. Each language model for a given document representation was estimated using a linear interpolation with the collection document representation model. The linear interpolation weights for the document representation models were set using Dirichlet priors as described in Section 2.2, with the prior parameter set to approximately twice the average document representation length. The parameters we used for Okapi BM25 are k1=1.2 and b=0.25.

Six different document representations were tested in our experiments: (i) the full document text, (ii) in-link text, (iii) title text, (iv) image alternate text, (v) large font text (including headings), and (vi) meta tag keywords and descriptions.

This is a subset of the document representations we used in [3]. Tables 2 and 3 summarize their performance using Okapi and Dirichlet prior smoothed language models. The best 3 document representations for both tasks and systems are the full text, in-link text, and the title text. These tables illustrate that some of the document representations tend to be better than other representations, independent of the retrieval system used.

4. EXPERIMENTAL RESULTS

This section presents experiments on combining document representations. We combined representations in order of their individual performance, using the top 100 results from each result list for the meta-search algorithms. The parameters used with the weighted algorithms were according to the document representations' individual performance on the training data when measured by MRR. As in previous work [1][13][19], we do not claim that these weights are optimal. This same method was used for choosing the linear interpolation parameters for the weighted version of the language model approach.

4.1 Compatibility Hypothesis

We begin our examination of the document representation score compatibility hypothesis, by presenting experiments with variants of the combMNZ algorithm. In Section 2.1 we mentioned two forms of score normalization: a linear scaling, which is standard, and transforming the original score using the exponential function. Figures 1a-d show the effects of these normalizations on meta-search using Okapi and language models.

For language models on the Homepage Finding task (Figure 1b), we found that normalizing the scores with the exponential function helped significantly. This is not surprising; Lemur returns the log of the probability, and applying the exponential function returns the score to the probability scale. Normalizing these scores using a linear transformation hurt performance. This may be because, in some sense, the generation probabilities are already normalized. We found similar results for Named-Page Finding (Figure 1d).

For combining Okapi scores on both tasks, we found that using the exponential to normalize the scores also helped performance. Perhaps this normalization helped due to the log-like functions Okapi uses in its ranking formula. It does not place the scores on a probability scale, as with the language models, but there may be some similar effect. We also tried the logistic regression transformation to the probability scale presented in [17], but it was not effective. We believe this may be due to the small number of positive examples for the training topics.

We hypothesize two representations are compatible (i.e. combine well) when they produce score distributions that have similar shapes. In order to measure this, we computed the mean-squared error (MSE) of document representation pairs for each query. We then averaged the MSE across queries and document representation pairs. In order to get comparable MSEs across

Homepage Finding

3 representations		6 representations	
Okapi (exp, not norm)	0.021	Okapi (exp, not norm)	0.013
LM (exp, not norm)	0.049	LM (exp, not norm)	0.038
Okapi (exp)	0.077	LM (exp)	0.088
LM (exp)	0.085	Okapi (exp)	0.090
Okapi	0.140	Okapi (not norm)	0.116
LM	0.160	Okapi	0.129
Okapi (not norm)	0.193	LM	0.148
LM (not norm)	0.335	LM (not norm)	0.215

Named-Page Finding

3 representations		6 representations	
Okapi (exp, not norm)	0.021	Okapi (exp, not norm)	0.013
LM (exp, not norm)	0.030	LM (exp, not norm)	0.019
LM (exp)	0.053	Okapi (exp)	0.054
Okapi (exp)	0.056	LM (exp)	0.059
LM	0.110	Okapi	0.094
Okapi	0.112	LM	0.103
Okapi (not norm)	0.200	Okapi (not norm)	0.106
LM (not norm)	0.399	LM (not norm)	0.218

Table 4: Scaled mean-squared error of the curves provided by the document representations averaged across topics and representation pairs.

experiments, we normalized the scores to range from zero to one using a linear transformation. This transformation was specific to any given query, ranking algorithm, and normalization method. This transformation was done *after* any normalization method applied to the rankings had been performed (for example, the linear transformation was done after the exponential normalization of scores). The MSE was computed over results at ranks one through thirty. Table 4 contains these numbers for both tasks when combining the best three and all six representations.

Within a retrieval algorithm the orderings of normalization techniques correspond exactly to the performance shown in Figures 1a-d. Across retrieval methods, we found that the orders were not exact predictors of performance, although similar performance does correspond to similar MSE. We attribute the differences in part to the fact that the MSE does not take into consideration the quality of the representations.

From Table 4 we can see that the MSEs from six representations are lower than the MSEs from three representations. The lower MSE scores found for all six representations do not mean that the results are more compatible. As more representations are added, it is likely that some of the curves will be near others, which will reduce the MSE. The MSE scores are only comparable where the same set of document representations are combined.

We now extend the examination of the score compatibility hypothesis to additional algorithms. Figures 2 and 3 show the results of using the unweighted and weighted algorithms on Homepage Finding task. Figure 4 shows results on the Named-Page Finding task. The unweighted algorithms only are used for Named-Page Finding, as there is no training data.

By our definition of compatability, using system ranks instead of scores gives perfect compatibility. However, the rank-based approaches do not perform as well as score-based aproaches. We believe that this is because by disregarding scores and using ranks, important information encoded in the probability estimates returned by the ranking algorithms is lost.

We omitted graphs showing the numbers of correct pages found by rank 10 in order to save space. The trends are largely the same as with mean-reciprocal rank, except that the relative performance of Borda count fusing Okapi results is slightly better for both the weighted and unweighted versions. Additionally, when fusing language models, the reciprocal rank fusion strategy's performance is better relative to the other systems for both the weighted and unweighted versions, and weighted Condorcet fuse's performance under found by rank 10 was better than its performance when considering mean-reciprocal rank. The relative system performances for the named-finding task when considering number of correct pages found by rank 10 was very similar to the mean-reciprocal rank measure. These differences do not change the ordering of the systems.

As in meta-search, we have demonstrated that compatibility is important for combining document representations. However, combining document representations is distinct from meta-search in that we can choose the representations being combined and the ranking algorithms used on the representations. With the added constraint that the ranking algorithms be the same for all representations, we can assume greater compatibility of the scores produced by the algorithms. Performing additional normalization of the scores can further improve the success of fusion.

4.2 Quality Hypothesis

This section investigates the hypothesis that the individual document representations must perform well in general to increase performance when fusing results. The best performing algorithms did not gain from including the three poorly performing representations (Figures 2-4). However, we sometimes find slight gains, and the best algorithms were robust to the addition of the other representations.

To investigate this further, we combined the three worst performing language model representations: image alternate text, large font text, and meta tag contents. Combining these representations using weighted combMNZ yielded a MRR of .303 on the Homepage Finding task and a MRR of .371 on the Named-Page Finding task. These results provide very strong evidence that the combined performance is better than any of the three individual document representations. Using the signed-rank test with correction for multiple testing using the Bonferroni method, the p-values were well under 0.001 for all comparisons to any of the original three representations. It is not clear from these results what the conditions are for having a weak document representation improve performance significantly; it is probably dependent on the document representations being combined.

In meta-search, much of the previous literature found improvements from combining search systems. However, a recent study suggests that there is not always a consistent improvement when combining result lists [2]. The authors hypothesize that when combining results with already high quality results, there is not much improvement, but when combining lower performance search engines, there is a more consistent improvement in the quality of the result lists. The findings in [2] are similar to ours.

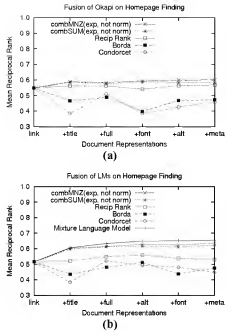


Figure 2: Results for Homepage Finding. The fusion algorithms are unweighted. (a) Okapi (b) language models

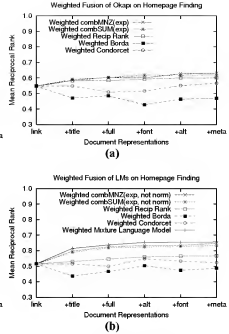


Figure 3: Results for Homepage Finding. The fusion algorithms are weighted. (a) Okapi (b) language models

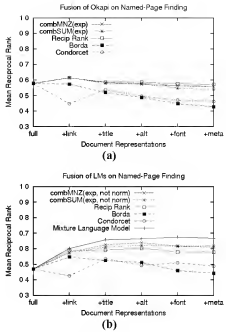


Figure 4: Results for Named-Page Finding. The fusion algorithms are unweighted. (a) Okapi (b) language models

To summarize, we found that the best algorithms were robust to poorly performing document representations, even if they could not leverage the weak representations. We also showed that there exist conditions under which poorly performing document representations can be combined to significantly improve performance. These results are consistent with recent results in meta-search. The quality hypothesis applies to both meta-search and combining document representations.

4.3 Variance Hypothesis

The third hypothesis was that in order to have successful fusion, the scores or ranks of the correct documents must vary less than those of the incorrect documents. This is not as easy to measure as the independence hypothesis made in meta-search. In meta-search it is appropriate to measure the overlap of the result lists for relevant documents and non-relevant documents. In our task, there is only one correct document and the document representations vary widely, it is not likely that the correct document will be returned by all of the search systems.

Table 5: Percentage of times the variance of the correct document was higher than the variance of other high ranking documents. The order of document representations combined is the same as in previous experiments. HP = homepage finding, NP = named-page finding, Okp = Okapi, LM = language models.

Task	Alg	Type	Number Representations				
			2	3	4	5	6
HP	Okp	Rank	62.5	73.1	81.9	88.1	90.9
		Score	86.3	89.0	90.4	90.6	92.3
	LM	Rank	60.8	71.0	80.8	85.9	89.7
		Score	90.8	93.3	93.3	92.4	93.7
NP	Okp	Rank	53.6	64.8	80.8	90.2	92.3
		Score	84.8	87.5	90.5	92.4	92.4
	LM	Rank	54.7	63.7	79.5	88.3	90.8
		Score	99.2	99.2	99.3	99.5	99.6

To investigate this hypothesis, we report the variance of the ranks/scores returned across all document representations of the correct document to the variance of results of the top 10 of any of the document representations. For a given query, the variance we measured is the variance of the document's rank/score across all document representations being combined. Table 5 presents the percentage of documents where the variance of the top incorrect documents was lower than the variance of the correct document.

The variance of the scores and ranks of the correct document across the representations was higher than for other highly ranked documents. This behavior was similar for the various ranking algorithms. Low variance of scores and ranks is not a factor in combining results lists formed from different representations.

It may not be surprising that the variance is high for the correct documents. In some document representations, the correct document may match the query very well, but not in all representations. For incorrect documents, none of the representations may match the queries well, which may yield scores that do not vary as much. This behavior distinguishes the problem of combining representations from the general meta-search problem. In meta-search, larger agreement among the scores of relevant documents than among the scores of irrelevant documents has been considered important for successful fusion. However, we find that low variance in scores of correct documents is not needed for successful for combination of representations.

Table 6: Statistical significance tests for combining language models on the .GOV named-page finding task

	mixture LM	combMNZ	combSUM	Recip rank	Borda fuse	Condorcet
Full text	<<	<	<	~	~	~
mixture LM		~	~	>>	>>	>>
combMNZ			~	~	>>	>>
combSUM				~	>>	>>
Recip rank					>>	>>
Borda fuse						~

4.4 Direct Comparisons/Significance Tests

It is easy to identify trends from these Figures 2-4. The score-based algorithms perform better than the ranked based algorithms. The language models representations seem to do better than the Okapi representations. Also, the mixture of language models performs best on all tasks. It is not clear whether these differences are significant.

Table 6 reports significance tests for combining 6 language model representations on .GOV Named-Page Finding. We used the Wilcoxon signed-rank test and corrected for multiple testing using the Bonferroni method. A > indicates that the algorithm for the row outperformed (according to MRR) the algorithm in the column with a p-value of 0.05 or less, while a >> indicates a p-value of 0.01 or less. A ~ indicates a p-value greater than 0.05. The < and << signs indicate the column algorithm out-performed the row algorithm.

There is very strong evidence the mixture-based language modeling approach is better than using the single best representation or using combining rank-based information. Additionally, there is very strong evidence that the score-based meta-search algorithms perform better than the Borda and Condorcet fusion algorithms.

We would also like to answer is whether the results from combining language models using meta-search algorithms are significantly better than the results from combining Okapi scores. The signed-rank test was performed across all system pairs, using the unweighted algorithms on both tasks. There was no statistical evidence of differences between language models or Okapi for the score based fusion algorithms.

However, we did find evidence that the mixture language model performed significantly better than the meta-search algorithms applied Okapi results. A p-value of 0.06 was obtained when comparing to combMNZ, 0.04 for combSUM, and under 0.01 for other approaches.

4.5 Comparison to TREC Results

For completeness, this section contains a brief comparison to results presented during and after the TREC conference by research other groups. For Homepage Finding, our results using the mixture language modeling system yielded an MRR of 0.658, failing to find only 5.5% in the top 100 returned results. For 82.8% of the topics, the correct document was in the top 10 results. We did not enter a submission to the Homepage Finding task, but this performance is among the best three groups that participated (out of 16). All of the submissions with higher performance made use of the URL text. In particular, Kraaij et al. [9] used a prior based on the depth of the URL. Incorporating their prior into our results increases the performance of the mixture language model to a MRR of 0.777 and finds a correct document in the top 10 results for 91% of the topics. This performance is comparable to that of Kraaij et al. [9] (MRR=0.774, found by rank 10=88.3%).

Our group did participate in the Named-Page Finding task of TREC 11. Our official submission used the same system, but included an additional language model created using the URL text. With this additional model, our system had a mean-reciprocal rank of 0.676. This was the second best official submission from a unique group. Without the URL model, the language model approach gives a mean-reciprocal rank of 0.667. Zhang et al. [22] had a mean reciprocal rank of 0.719 for their best submission. Zhang et al. use similar structural information, along with creating separate indexes for html and pdf documents. Park et al. [15] also report comparable results. Their work uses a query-sentence similarity measures in addition to query-document similarity. Both groups used link anchor text in addition to other information present in the document.

5. CONCLUSIONS

In this paper, we explored the use of meta-search algorithms and a language modeling approach to the combination of document representations. The experiments were carried out on two known-item finding tasks on HTML documents: Homepage Finding and Named-Page Finding.

We investigated three hypotheses adapted from the task of meta-search. The first hypothesis was that the outputs from the search systems have compatible output. We found this to be very important. We proposed a measure of score compatibility using mean-squared error that accurately ordered the score normalization techniques according to their effectiveness. We found that system scores could be leveraged effectively, and that rank-based fusion algorithms did not perform as well as the score-based algorithms. A mixture-based language model gave better results than applying meta-search techniques to Okapi results. We additionally found that normalizing Okapi scores by applying the exponential function improved the compatibility of the scores.

Our second hypothesis was that for successful performance in combining document representations, each system should perform well. We demonstrated that this hypothesis is not true; document representations that perform poorly can be combined with other representations to improve performance. However, this gain in performance is not guaranteed, and adding representations to a high performing system may not improve performance. We also demonstrated that the best algorithms for combining representations tend to be robust to the addition of representations. That is, including new representations rarely hurt the performance. When the new document representations did decrease performance, the degradation was not significant.

The third hypothesis was that for successful combination of document representations, the scores or ranks of the correct documents across document representations would vary less than those of incorrect documents. We found this to be false. The scores and ranks of correct documents varied more than those of

the incorrect documents, possibly because the correct documents would match the query very well in some of the document representations, but not well in others.

The best meta-search algorithms did not perform quite as well as the mixture-based language model. While these differences were not large or significant, this lends merit to the approach of directly combining the document representations within a retrieval model. We found that the combMNZ algorithm worked best among the meta-search algorithms. We demonstrated that the weighted versions of all algorithms could improve performance, even when using a non-optimal training strategy for selecting the weights.

The problem of combining document representations is distinct from the meta-search problem. When combining document representations, one can choose which ranking algorithms are used. This added flexibility opens the doors for new methods of evidence combination and more appropriate normalization methods tailored to the ranking algorithms. Combining document representations is also different from meta-search in that the success of the combination methods is not dependent on the variance of scores or ranks of correct documents being lower than those of the incorrect documents.

Combining representations is an old idea within Information Retrieval, but the emergence of the Web and XML give it new importance. Structured documents are becoming more common, not less, giving new urgency to developing retrieval models that manage them effectively. The work reported here suggests that statistical language models are a sound foundation on which to construct such systems.

6. ACKNOWLEDGMENTS

This research was supported by the Advanced Research and Development Activity in Information Technology (ARDA) under its Statistical Language Modeling for Information Retrieval Research program, and by NSF grant EIA-9983253. Any opinions, findings, conclusions, or recommendations expressed in this paper are the authors, and do not necessarily reflect those of the sponsors.

7. REFERENCES

[1] J.A. Aslam and M. Montague. Models for Metasearch. In Proc. of the 24th Annual International ACM SIGIR Conf. on Research and Development in Information Retrieval, pages 276-284, 2001. ACM.

[2] A. Chowdhury, O. Frieder, D. Grossman, and C. McCabe. Analyses of multiple-evidence combinations for retrieval strategies. In Proc. of the 24th Annual International ACM SIGIR Conf. on Research and Development in Information Retrieval, pages 394-395, 2001. ACM.

[3] K. Collins-Thompson, P. Ogilvie, Y. Zhang, and J. Callan. Information filtering, novelty detection, and named-page finding. In Proceedings of the 11th Text REtrieval Conference (TREC-11), pages 338-349, notebook version, 2002.

[4] N. Craswell, D. Hawking, and S. Robertson. Effective site finding using link anchor information. In Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 250-257, 2001. ACM.

[5] W.B. Croft. Combining approaches to information retrieval. In Advances in Information Retrieval: Recent Research from the Center for Intelligent Information Retrieval, chapter 1, pages 1-36. Kluwer Academic Publishers, 2000.

[6] E.A. Fox and J.A. Shaw. Combination of multiple searches. In The Second Text REtrieval Conference (TREC-2), pages 243-249, 1994.

[7] N. Fuhr, N. Gövert, G. Kazai, and M. Lalmas, editors. INEX 2002 Workshop Proceedings. To be published. Draft available at http://qmir.dcs.qmul.ac.uk/inex/Workshop.html.

[8] D. Hawking and N. Craswell. Overview of the TREC-2001 Web Track. In Proceedings of the 10th Text REtrieval Conference (TREC-10), pages 61-67, 2002.

[9] W. Kraaij, T. Westerveld, D. Hiemstra. The importance of prior probabilities for entry page search. In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 27-34, 2002. ACM.

[10] The Lemur toolkit for language modeling in information retrieval. http://www.cs.cmu.edu/~lemur

[11] S.H. Myaeng, D.H. Jang, M.S. Kim, and Z.C. Zhoo. A flexible model for retrieval of SGML documents. In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 138-145, 1998. ACM.

[12] R. Manmatha, T. Rath, and F. Feng. Modeling score distributions for combining the outputs of search engines. In Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 267-275, 2001. ACM.

[13] M. Montague and J.A. Aslam. Condorcet fusion for improved retrieval. In Proc. of the 11th International Conf. on Information Knowledge and Management (CIKM), pages 538-548, 2002. ACM.

[14] K.B. Ng and P. Kantor. An investigation of the preconditions for effective data fusion in IR: a pilot study. In Proc. of the 61st Annual Meeting of the American Society for Information Science, 1998.

[15] E.K. Park, S.I. Moon, D.Y. Ra, and M.G. Jang. Web Document Retrieval Using Sentence-query Similarity. In Proceedings of the 11th Text REtrieval Conference (TREC-11), notebook version, 2002.

[16] J. Ponte and W.B. Croft. A language modeling approach to information retrieval. In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 275-281, 1998. ACM.

[17] S.E. Robertson. Threshold Setting and Performance Optimization in Adaptive Filtering. In Information Retrieval, volume 5(2-3), pages 239-256. Kluwer Academic Publishers, 2002.

[18] J. Savoy, A.L. Calvé, and D. Vrajitoru. Report on the TREC-5 experiment: data fusion and collection fusion. In The 5th Text REtrieval Conference (TREC-5), pages 489-502, 1997.

[19] C.C. Vogt and G.W. Cottrell. Fusion via a linear combination of scores. Information Retrieval, 1(3), pages 151-173, Oct. 1999.

[20] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In Proc. of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 334-342, 2001. ACM.

[21] C. Zhai and J. Lafferty. Two-Stage Language Models for Information Retrieval. In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 49-56, 2002. ACM.

[22] M. Zhang, R. Song, C. Lin, L. Ma, Z. Jiang, Y. Jin, Y. Liu, L. Zhao, and S. Ma. THU at TREC 2002: novelty, web, and filtering (draft). In Proceedings of the 11th Text REtrieval Conference (TREC-11), pages 29-42, notebook version, 2002.